

NI Developer Zone

Document Type: [Tutorial](#)
NI Supported: Yes
Publish Date: Jan 19, 2007

Creating a Custom Component in NI Multisim

Overview

NI Multisim and **NI Ultiboard** provide an integrated platform to design, simulate and lay out a complete Printed Circuit Board (PCB). The highly flexible database manager makes it easy to add a new SPICE simulation model, to a custom defined schematic symbol which can then transfer an accurate footprint to layout.

[Creating a Custom Component in NI Multisim](#) and [Creating a Custom Component in NI Ultiboard](#) are your resource on how to intuitively and quickly learn how to create your own custom components.

Table of Contents

- [1. Introduction](#)
- [2. Step 1: Enter Initial Component Information](#)
- [3. Step 2: Enter Footprint and Package Information](#)
- [4. Step 3 -- Enter Symbol Information](#)
- [5. Step 4: Set Pin Parameters](#)
- [6. Step 5: Set Mapping Information between Symbol and Layout Footprint](#)
- [7. Step 6: Select the Simulation Model](#)
- [8. Step 7: Map Symbol Pins to the Model Nodes](#)
- [9. Step 8: Save Component into the Database](#)
- [10. Step 9 -- Testing the New Component in Multisim](#)

Introduction

This tutorial is the first in a series of articles on component creation in **NI Multisim** and **NI Ultiboard**.

The purpose of this tutorial is to explain how you can create your own components for simulation and/or Printed Circuit Board (PCB) layout from within Multisim. You will create the component and then verify its operation. The Component Wizard is the primary tool used to create custom components, and it walks you through all of the steps required to create a new component. Components details include symbol and optionally pin, model, and footprint information. Steps in the creation process include:

- Entering component information
- Selecting a footprint and the component configuration
- Selecting and/or editing the component symbol
- Setting pin parameters
- Mapping the symbol pins to the footprint pins
- Selecting the simulation model
- Mapping the symbol pins to the model pins
- Saving the part in the database

The tutorial steps you through the process of creating a simulation and PCB layout compatible component. In the interest of completeness, you will learn how to create an advanced component with two sections. You will create a part with two schematic symbols, two models, but only one footprint. Many components are easier to create and not all steps outlined are required under most circumstances. Multisim also allows users to create simulation-only or layout-only components.

Part 2 of the component creation articles, entitled [Creating a Custom Component in NI Ultiboard](#), outlines how to create a custom Ultiboard landpattern for layout. This landpattern is created manually so as to precisely define the shape, size, and dimensions of a Surface Mount Device (SMD). This footprint can be added to the Multisim database to define a custom component.

Single Section versus Multisection Components

A single-section component is any component that has a single device per chip. A multisection component is a component that has multiple gates or devices per chip. Examples of multisection devices include logic gates or operational amplifiers. Letters, increasing from A-Z, enumerate the devices within multi-section components.

An example of a multi-section component is the Texas Instruments® THS7001. The THS7001 is a programmable gain

amplifier (PGA) with a separate pre-amplification stage presented in a single integrated circuit (IC) package. In this package, both sections share power and reference voltage connections. You will learn how to create this component in this tutorial.

Simulation-Only Components

Simulation-only components are designed to help verify designs; they are not transferred to board layout. There is no footprint information associated with them, and their symbols are colored black by default in the Multisim or Multicap environment to easily identify them. An example of a Simulation-only component is an ideal voltage source.

Layout-Only Components

Layout-only components do not contribute to a simulation. They have no associated SPICE, VHDL, or behavior model. When connected parallel to the circuit, they will have no impact on the simulation. When connected in series, they will create an open circuit. Layout-only components are green on the Multisim or Multicap environment. An example of a Layout-only component is a connector.

Creating a Texas Instruments® THS7001 Component in NI Multisim

The Texas Instruments® THS7001 is a programmable gain amplifier with a separate pre-amplification stage. The programmable gain is digitally controlled via three TTL-compatible inputs. The datasheet for the THS7001 is included below in Appendix A for reference.

Step 1: Enter Initial Component Information

Start the Component Wizard by selecting **Tools » Component Wizard** from the Multisim Main Menu. From this window, enter the initial component information (Figure 1). Select the **Component Type** and use-case (simulation, layout, or both).

Choose **Next >** when done.

Component Wizard - Step 1 of 8

Enter Component Information

Component Name: THS7001

Author Name: Electronics Workbench

Component Type: Analog

Function: 70 MHz Programmable Gain Amplifier (PGA) with separate preamp and PGA stages.

I will use this component for both simulation and layout (model and footprint)

Simulation only (model)

Layout only (footprint)

Next > Cancel Help

Figure 1 - THS7001 Component Information

Step 2: Enter Footprint and Package Information

a) Click **Select a Footprint** to choose a footprint for this component.

Note: When creating a simulation-only component, the footprint information fields are grayed out.



Figure 2 - Select a Footprint (1 of 2)

b.) Select the footprint listed in the manufacturer datasheet. For the THS7001, select the TSSOP20 from the Master Database. Choose **Select** when done.

*Note: If you know the name of the footprint, you can also type it directly into the **Footprint Type** field.*



Figure 3 - Select a Footprint (2 of 2)

c.) Define the names of the sections and number of pins for each of the sections in the component. In this case, there are two sections: A is the pre-amplifier section, and B is the programmable gain amplifier section.

Note 1: When creating multi-section components, the number of pins must match the number of pins that will be used for that section's symbol not the number of pins of the footprint.

Note 2: In the case of the THS7001, you will add the ground pin and the power-saving shut-down pin to the symbols for both sections.

Choose **Next** when done.

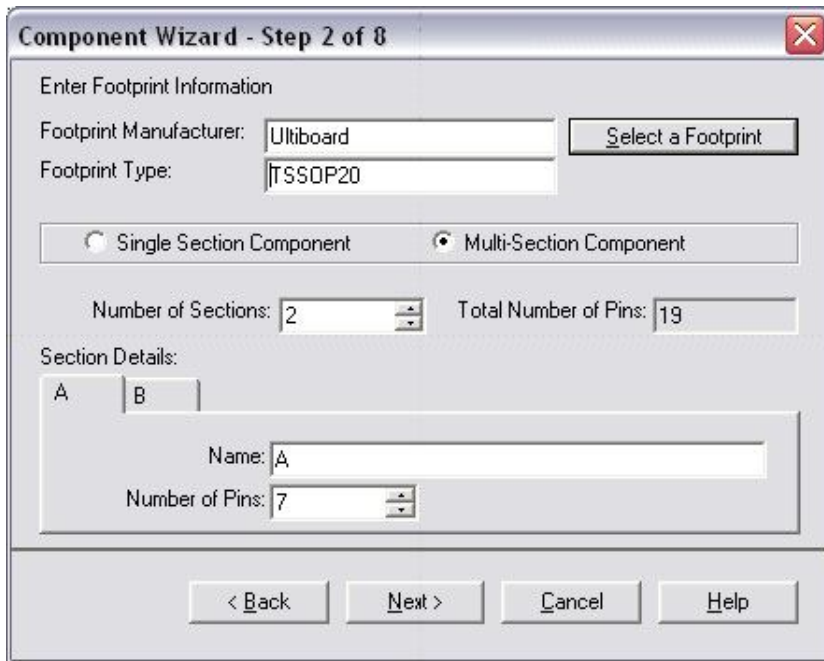


Figure 4 - Defining a Multisection Part (1 of 2)



Figure 5 - Defining a Multisection Part (2 of 2)

Note: To learn how to create a custom footprint in NI Ultiboard take a look at the [Creating a Custom Component in NI Ultiboard](#).

Step 3 -- Enter Symbol Information

After defining the sections and selecting the footprint, assign the symbol information for each section. You can assign symbols by editing them in the Symbol Editor (choose **Edit**) or copying existing ones from the database (choose **Copy from DB**). To save time when creating custom parts, it is recommended that you copy existing symbols from the database wherever possible. You can also load symbol files into the Symbol Editor. The symbols for the THS7001 are included as files with this tutorial.

a.) Load the symbol for the preamp device:

Choose **Edit** to open the symbol editor.

Once the symbol editor has loaded, select **File » Open** and navigate to where you saved the tutorial files. Select **preamp.sym**. The loaded symbol is shown below in Figure 6.

Note 1: With the exception of the common shut-down and ground pins, the names of the pins have been prefixed with "PA". This is to differentiate the names of the pins on the preamp from the names of pins on the programmable gain amplifier.

Note 2: In order for shared pins to correctly function in the capture environment, they must have the same name across sections. Additionally, they must be assigned to the COM (common) section in step 4.

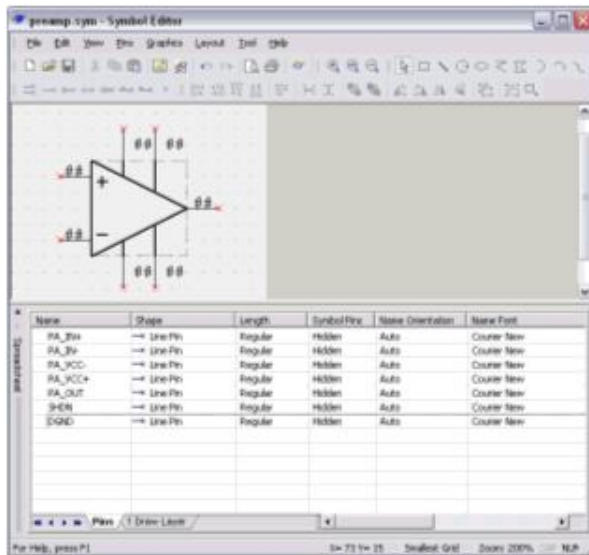


Figure 6 - Preamplifier Symbol

Close the symbol editor. If asked to save, choose **Yes**.

The preamp symbol will now be displayed in the preview box. If you plan to share this component with colleagues around the world, it is a good idea to create both ANSI and DIN symbols for the device. In this case, simply choose **Copy to ...** then select the only visible choice of either **Section A (ANSI)** or **Section A (DIN)**.

b.) Load the symbol for the PGA.

Select device B and choose **Edit** to launch the Symbol Editor.

Choose **File » Open** and navigate to where you saved the tutorial files and select **pga.sym**. The resulting symbol is shown below in Figure 7.

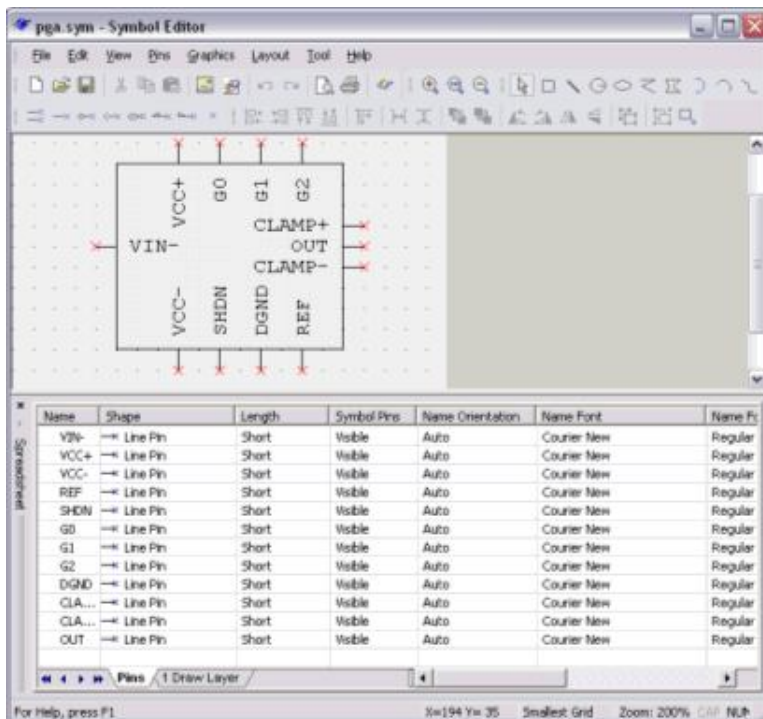


Figure 7 - Programmable Gain Amplifier Symbol

Close the symbol editor. If asked to save, choose **Yes**.

Note: If at this time the Multisim window does not re-appear, follow the note listed in the troubleshooting section in Appendix B.

The PGA symbol is shown in the preview box. If you plan to share this component with colleagues around the world, it is a good idea to create both ANSI and DIN symbols for the device. In this case, simply choose **Copy to ...** then select the only visible choice of either **Section A (ANSI)** or **Section A (DIN)**.

Choose **Next** when done.

Step 4: Set Pin Parameters

All the pins for this component are listed in Step 4 and are illustrated Figure 8 below. Multisim uses the pin parameters when running an electrical rules check. The pin parameters are also required to select the correct pin drivers for digital components. You can also add hidden pins to components in this step. Hidden pins are pins that do not appear on the symbol but may be used by the model and/or the footprint.

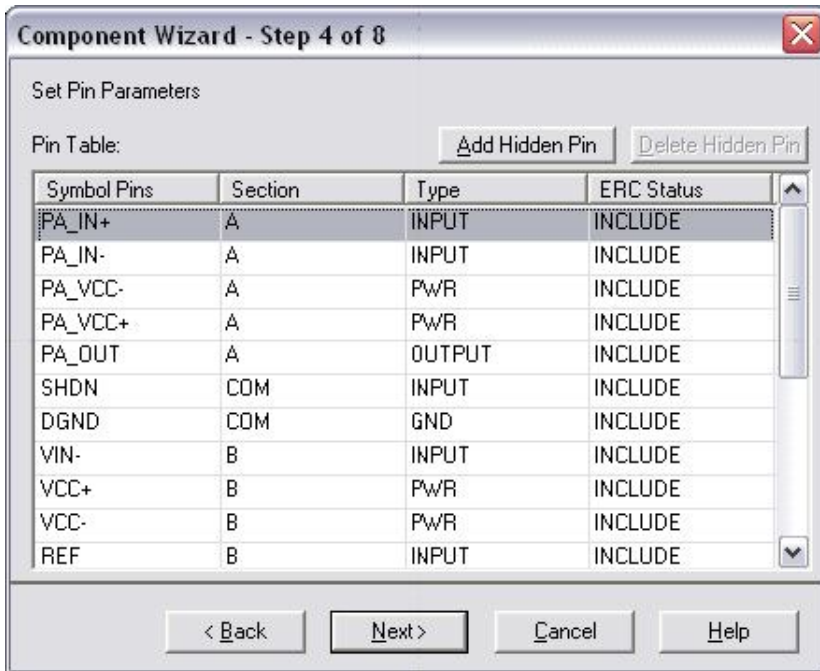


Figure 8 - Pin Parameters

a.) Complete the pin table as shown below in Table 1.

Symbol Pins	Section	Type	ERC Status
PA_IN+	A	INPUT	INCLUDE
PA_IN-	A	INPUT	INCLUDE
PA_VCC-	A	PWR	INCLUDE
PA_VCC+	A	PWR	INCLUDE
PA_OUT	A	OUTPUT	INCLUDE
SHDN	COM	INPUT	INCLUDE
DGND	COM	GND	INCLUDE
VIN-	B	INPUT	INCLUDE
VCC+	B	PWR	INCLUDE
VCC-	B	PWR	INCLUDE
REF	B	INPUT	INCLUDE
SHDN	COM	INPUT	INCLUDE
G0	B	INPUT	INCLUDE
G1	B	INPUT	INCLUDE
G2	B	INPUT	INCLUDE
DGND	COM	GND	INCLUDE
CLAMP-	B	INPUT	INCLUDE
CLAMP+	B	INPUT	INCLUDE
OUT	B	OUTPUT	INCLUDE

Table 1 - THS7001 Pin Parameters

Choose **Next** when done.

Step 5: Set Mapping Information between Symbol and Layout Footprint

Map visible symbol pins and hidden pins to the PCB footprint in Step 5. Simulation-only parts do not require this step.

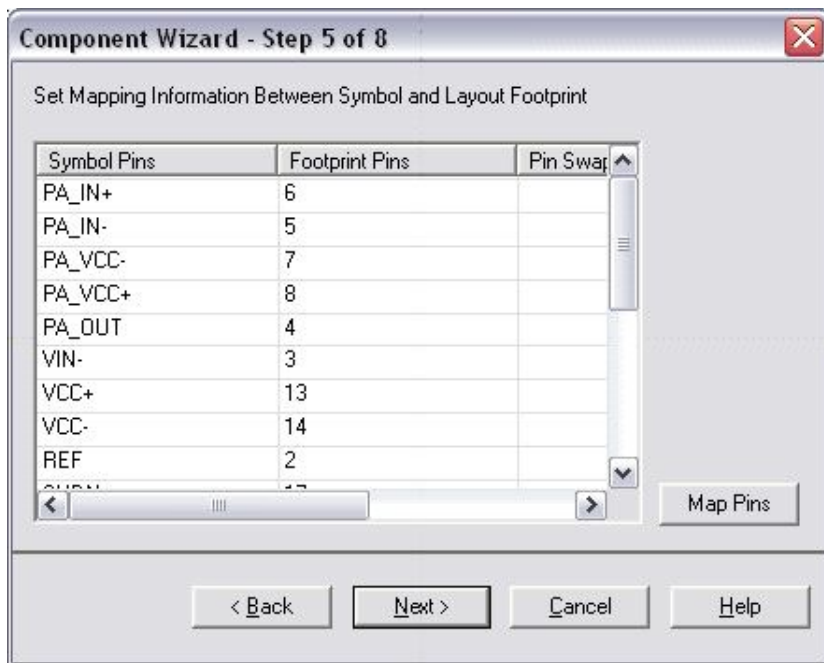


Figure 9 - Symbol to Footprint Mappings

a.) Use the datasheet as reference to complete the mapping information shown below in Table 2.

Note: Pin 17 is shared by SHDN and PA_SHDN, and pin 1 is shared by DGND and PA_GND.

Symbol Pins	Footprint Pins	Pin Swap Group	Gate Swap Group
PA_IN+	6		
PA_IN-	5		
PA_VCC-	7		
PA_VCC+	8		
PA_OUT	4		
SHDN	17		
DGND	1		
VIN-	3		
VCC+	13		
VCC-	14		
REF	2		
SHDN	17		
G0	20		
G1	19		
G2	18		
DGND	1		
CLAMP-	15		
CLAMP+	12		
OUT	16		

Table 2 - Symbol to Footprint Mapping

Choose **Next** when done.

Note 1: Pins that belong to the same Pin Swap group can be interchanged automatically in board layout to maximize routing efficiency. Often chips will have several ground pins. Assign these to a Pin Swap group, and the Ultiboard PCB Layout tool will annotate the net list in order to improve the physical layout of the board.

Note 2: Additionally, some chips will have multiple devices of the same type (the 74HC00 contains 4 identical digital NAND gates). For improved routing, these gates can be assigned to a gate swap group.

No two pins on the PCB footprint of the THS7001 are repeated. Likewise there are no two identical gates. Therefore, the pin and gate swap information is left blank.

Step 6: Select the Simulation Model

When creating a component for simulation, you must provide the simulation models for each section. You can obtain or create new models in 4 ways:

- Download a SPICE model from a manufacturer website or other source
- Manually create a subcircuit or primitive model
- Use a Multisim Model Maker
- Or edit an existing model

Multisim provides Model Makers that create SPICE models for several categories of components based on their databook values. Model Makers exist for operational amplifiers, bipolar junction transistors, diodes, waveguides, and many others. For more information on the various model makers, consult the Multisim help file.

For the THS7001, you will use manufacture-provided SPICE compatible models. There are separate models for the preamp and PGA sections.

Note: If you are creating a layout-only part, you are not required to complete steps 6 and 7.

a.) With the tab for section A selected, choose **Load from File**. Navigate to the folder that contains the tutorial files, select **sloj028.cir** and choose **Open**. The SPICE model for the preamp will load and be displayed in the tab for section A as shown below.



Figure 10 - SPICE Model for THS7001 Preamp Stage

b.) Select the tab for section B, and choose **Load from File** to load the SPICE model for the PGA stage. Navigate to the folder that contains the tutorial files, select **sloj029.cir** and choose **Open**. The spice model is displayed in the B section of the Component Wizard step 6.



Figure 11 - SPICE Model for THS7001 PHA Stage

Choose **Next** when done.

Step 7: Map Symbol Pins to the Model Nodes

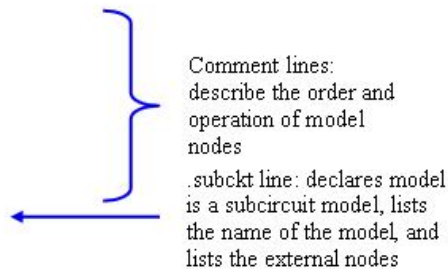
You must map the symbol pins to the SPICE model nodes in order for Multisim to correctly simulate the component.

For all subcircuit or macro models, the model nodes are typically documented in the header text of the SPICE model. There is also a line that declares that the model is a subcircuit model and lists the model name followed by the model nodes that will connect to external circuitry.

For the THS7001, the model nodes of the preamp are listed in **sloj028.cir** and the model nodes of the PGA are listed in **sloj029.cir**.

Let us examine the header and .SUBCKT lines for the preamp:

```
* THS7001 PREAMP SUBCIRCUIT REV -
* WRITTEN 5/3/00
* CONNECTIONS:  NON-INVERTING INPUT
*                | INVERTING INPUT
*                || POSITIVE POWER SUPPLY
*                ||| NEGATIVE POWER SUPPLY
*                |||| OUTPUT
*                ||||| SHUTDOWN
*                ||||| DGND
*                |||||
.SUBCKT THS7001_PRE 1 2 3 4 5 6 7
```



[+] Enlarge Image

You must now map the symbol pins names to the model nodes. **Pay special attention to the order of the model nodes.**

a.) Complete the pin mapping table for the preamp section A as shown below in Table 3.

Symbol Pins	Model Node Order
PA_IN+	1
PA_IN-	2
PA_VCC-	4
PA_VCC+	3
PA_OUT	5
SHDN	6
DGND	7

Table 3 - Symbol to Model Node Mapping for the Preamp

b.) Click on the tab for Section B and complete the pin mapping table for the PGA section B as shown below in Table 4.

Symbol Pins	Model Node Order
VIN-	2
VCC+	3
VCC-	4
REF	1
SHDN	6
G0	7
G1	8
G2	9
DGND	12
CLAMP-	11
CLAMP+	10
OUT	5

Table 4 - Symbol to Model Node Mapping for the PGA

Choose **Next** when done.

Step 8: Save Component into the Database

Once all the previous steps are complete, save the component to the Corporate or User databases.

a.) Select the database, group, and family to which you would like to save the component. If no family currently exists in the selected group, create a new family by choosing **Add Family**.

b.) Choose **Finish** to complete the component.

Note: You can customize the icon for a new family in the database manager by choosing **Tools » Database » Database Manager** from the Multisim main menu.



Figure 12 - Saving the Component to the Database

SUCCESS!!

Step 9 -- Testing the New Component in Multisim

After creating and saving the component, it is available for use in Multisim. To test this component, use **THS7001 Tester.ms9** file that is included in this tutorial. Replace U2a with section A of your new component and replace U2b with section B. To replace a component, double-click on the component and then select **Replace**. Then navigate to the database and location that you saved your component and select it. Select the appropriated section.

Figure 13 to Figure 16 below illustrate the expected responses of the test circuit.

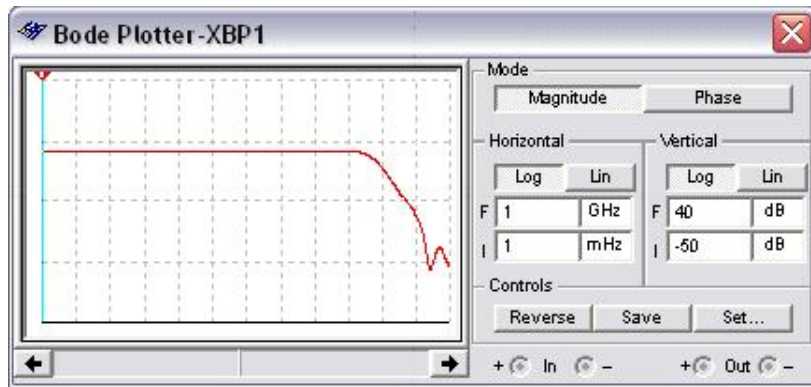


Figure 13 - Bode Response for the Preamp of the test Circuit

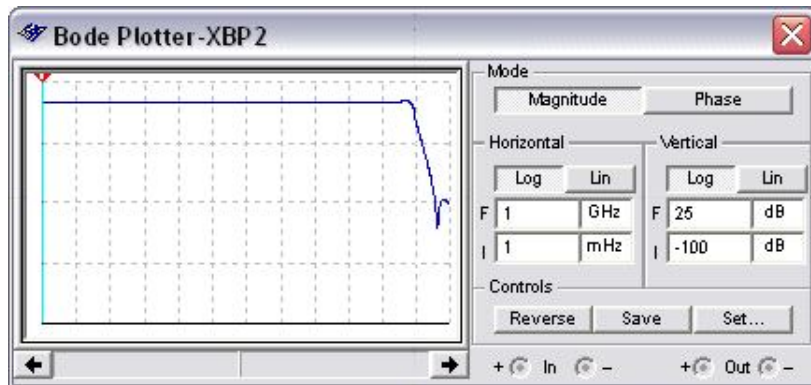


Figure 14 - Bode Response of the PGA with Gain Set to '111'

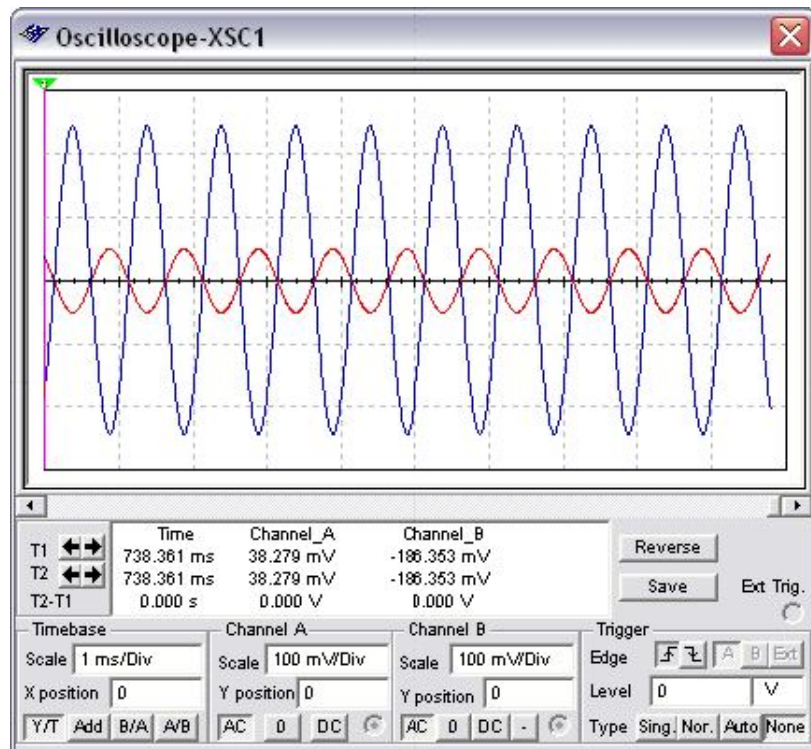


Figure 15 - Time Domain Response of the Preamp

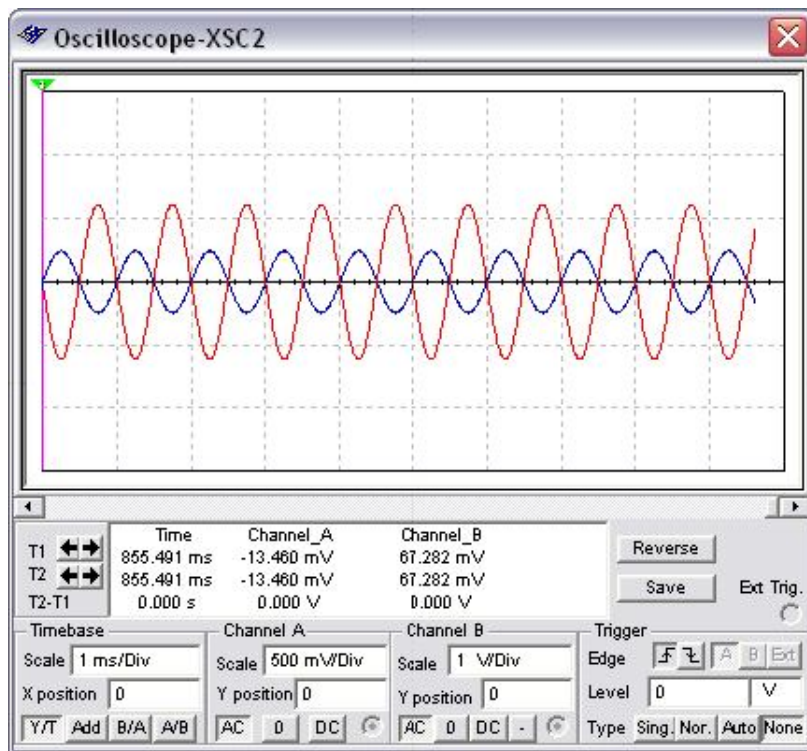


Figure 16 - Time Domain Response of the PGA with Gain set to '111'

Downloads

[ths7001.pdf](#)

[preamp.sym](#)

[ths7001testcircuit.ms9](#)

[sloj029.cir](#)

[pga.sym](#)

[sloj028.cir](#)

Reader Comments | [Submit a comment »](#)

no ths7001 tester.ms9 file

Step 9 -- Testing the New Component in Multisim states: """"""To test this component, use THS7001 Tester.ms9 file that is included in this tutorial."""""" i can not find this file in this tutorial. After creating and saving the component, it is available for use in Multisim. """"""To test this component, use THS7001 Tester.ms9 file that is included in this tutorial. """"""

Replace U2a with section A of your new component and replace U2b with section B. To replace a component, double-click on the component and then select Replace. Then navigate to the database and location that you saved your component and select it. Select the appropriated section.

- *Testing the New Component in Multisim, Power of Design.* super@powerofdesign - Dec 15, 2006

Legal

This tutorial (this "tutorial") was developed by National Instruments ("NI"). Although technical support of this tutorial may be made available by National Instruments, the content in this tutorial may not be completely tested and verified, and NI does not guarantee its quality in any way or that NI will continue to support this content with each new revision of related products and drivers. THIS TUTORIAL IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND AND SUBJECT TO CERTAIN RESTRICTIONS AS MORE SPECIFICALLY SET FORTH IN NI.COM'S TERMS OF USE (<http://ni.com/legal/termsofuse/unitedstates/us/>).